

ИМИТАЦИОННАЯ МОДЕЛЬ И ЕЕ ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ПЛАНИРОВАНИЯ РЕСУРСОВ ГРИД-СИСТЕМЫ

С.В. Минухин, С.В. Знахур

Харьковский национальный экономический университет, 61166, Харьков, проспект Ленина, 9-А,
тел.: (057) 702 1831. e-mail: ms_vl@mail.ru, sergznakhur@mail.ru

Рассмотрена модель и ее программная реализация для имитации процессов планирования ресурсов Грид-системы. Предложены метрики производительности системы и проведен сравнительный анализ разработанного метода планирования на основе задачи о наименьшем покрытии и метода FCFS. Приведены результаты вычислительных экспериментов, обосновывающие эффективность разработанного метода планирования.

A model and its software implementation to simulate the process of resource scheduling Grid-system are proposed. Metrics of system performance are proposed. A comparative analysis of the developed method based scheduling tasks on the minimal cover and the method of FCFS. The results of computational experiments which substantiate the effectiveness of the developed method of scheduling presented.

Введение

Современное развитие информационно-коммуникационных технологий все более трансформируется в направлении использования технологий распределенных и параллельных вычислений. Основной формой организации систем управления сложными вычислениями являются Грид-системы, которые нашли достаточно широкое применение в современных исследованиях в области физики, химии, биологии, проектировании сложных конструкций, медицине и здравоохранении. В этих системах используются архитектуры, обеспечивающие мониторинг состояния их компонент, обработку событий, связанных с поступлением и обработкой заданий пользователей, своевременную передачу результатов вычислений и множество других процессов, связанных с ее функционированием. Важной задачей при эксплуатации Грид-систем является обеспечение требуемой системной производительности при решении задач различной трудоемкости на имеющихся гетерогенных ресурсах, которая определяется на основе анализа таких характеристик, как производительность составляемых расписаний выполнения заданий на ресурсах, – среднее время выполнения одного задания, максимальное время завершения заданий, время ожидания в очереди, – а также характеристики, определяющие сбалансированность ресурсов (кластеров) Грид. Наряду с необходимостью расчета и анализа этих характеристик требуется постоянно отслеживать состояние ресурсов (узлов) системы на предмет их доступности и готовности для проведения вычислений. Наиболее актуальной эта задача является для интенсивных потоков заданий, которые могут выполняться в последовательном и параллельном режимах, использовать различное программное обеспечение и иметь различные требования к архитектуре и программному обеспечению ресурсов. Указанные задачи целесообразно решать на основе моделирования процессов функционирования Грид-систем, позволяющих разрабатывать сценарии управления системой в условиях динамично изменяющихся количества пользователей, характеристик заданий и ресурсов. В настоящее время существует достаточно большое количество программ и симуляторов, использующихся для моделирования Грид-систем [1 – 6], назначение и основные характеристики которых приведены в табл. 1, а также работ, посвященных их применению [7, 8].

Из табл. 1 следует, что достаточно высокая функциональность приведенных пакетов обеспечивает высокую эффективность моделирования и возможность анализа состояний объектов Грид-инфраструктуры, но, вместе с тем, они имеют определенные недостатки:

отсутствие необходимого для количественного и качественного анализа результатов планирования метрик;

сложность интеграции новых алгоритмов планирования в программную реализацию;

необходимость наличия достаточно полной и точной информации о характеристиках заданий и ресурсов;

отсутствие инструментов определения периодов планирования в соответствии с имеющимися данными;

отсутствие методов динамического планирования, позволяющих оперативно учитывать изменения в инфраструктуре Грид и интенсивностей входного потока заданий в процессе моделирования.

Цель данного исследования – разработка имитационной модели планирования ресурсов Грид-системы на основе пакетного планирования заданий, обеспечивающего высокий уровень загрузки вычислительных ресурсов, разработка ее программной реализации и сравнительный анализ результатов использования предлагаемого метода с существующими на основе выбранных метрик производительности Грид-системы.

Описание модели

В работе предлагается модель и ее программная реализация GRID_Scheduler_Model, включающая следующие функциональные компоненты, реализующие:

Таблица 1. Инструменты для моделирования параллельных и распределенных систем

© С.В. Минухин, С.В. Знахур, 2012

№	Название	Описание	Разработчик	Назначение
1	2	3	4	5
1	SimJava	Поддерживает процессы, основанные на дискретном моделировании событий с анимацией объектов через коллекцию сущностей, которые взаимодействуют друг с другом	Эдинбургский университет, Великобритания	Дискретно-событийное моделирование
2	Bricks	Инструмент для оценки, который анализирует и сравнивает различные схемы планирования в высокопроизводительной Грид-среде	Токийский технологический институт, Япония	Моделирование Грид
3	MicroGrid	Инструмент моделирования, который поддерживает масштабируемое моделирование Грид-приложений с использованием кластерных ресурсов в исследованиях вычислительных Грид	Калифорнийский университет, Сан-Диего, США	Моделирование Грид
4	SimGrid	Инструмент для моделирования работы вычислительного Грид в гетерогенных распределенных средах	Калифорнийский университет, Сан-Диего, США	Моделирование Грид
5	GridSim	Инструмент для моделирования сущностей, участвующих в параллельных и распределенных вычислениях путем создания и мониторинга пользователей и ресурсов	Университет Монаш, Австралия	Моделирование Грид
6	SimOS	Предоставляет полную среду моделирования вычислительных машин, имитирующую компьютерную технику для однопроцессорных и многопроцессорных вычислительных систем	Стэнфордский университет, США	Моделирование вычислительных ресурсов
7	OMNeT++	Компонентная модульная и открытая архитектура для дискретно-событийного моделирования	OpenSim Ltd. Венгрия	Моделирование компьютерных сетей, а также сетей массового обслуживания и других областях
8	GSSIM	Инструмент для планирования ресурсов на основе моделирования широкого спектра алгоритмов планирования с использованием брокеров и провайдеров ресурсов	Центр суперкомпьютерных и сетевых технологий, Технологический университет, Познань, Польша	Моделирование многоуровневых гетерогенных инфраструктур Грид
9	MPISim	Инструменты для моделирования параллельных программ с использованием Message Passing Interface (MPI)	Калифорнийский университет, Лос-Анджелес, США	Моделирование работы параллельных систем
10	Ptolemy 2	Java-библиотека, моделирование параллельных гетерогенных систем, работающих в режиме реального времени	Калифорнийский университет, Беркли, США	Моделирование работы параллельных систем
11	OptorSim	Моделирование на языке Java для имитации фактических структур данных, планирование работ и стратегий репликации данных	The DataGrid Project	Моделирование работ и репликаций данных

множество заданий – данные, которые подаются на систему для моделирования процесса работы системы;
 пул заданий – пакет заданий на входе системы, выбираемый определенным образом из множества заданий глобальной очереди;

методы формирования пакета заданий – модуль, который распределяет установленным методом задания, находящиеся в пуле между ресурсами, формирует пакет заданий на ресурс;

ресурси – елементи, симулюючі рішення завдань.

Перед моделюванням роботи системи формується множество завдань глобальної череди, которое будет подано на систему за период моделювання.

Процесс моделювання роботи системи включает пошаговое выполнение 4-х операций (событий).

Операция 1 – имитация поступления заданий на вход системы. Из заданий глобальной очереди в пул загружаются задания, выбранные по определенному правилу.

Операция 2 – распределение заданий из очереди пула на ресурсы и возврат неразмещенных заданий обратно в пул.

Операция 3 – имитация решения заданий на ресурсах. Данная операция использует в качестве единицы измерения времени имитации решения условную единицу времени – такт.

Операция 4. Переход к выполнению операции 1.

Все временные процессы при моделировании работы системы измеряются в тактах – времени модели системы. На рис. 1 показана схема алгоритма моделирования работы Грид-системы.

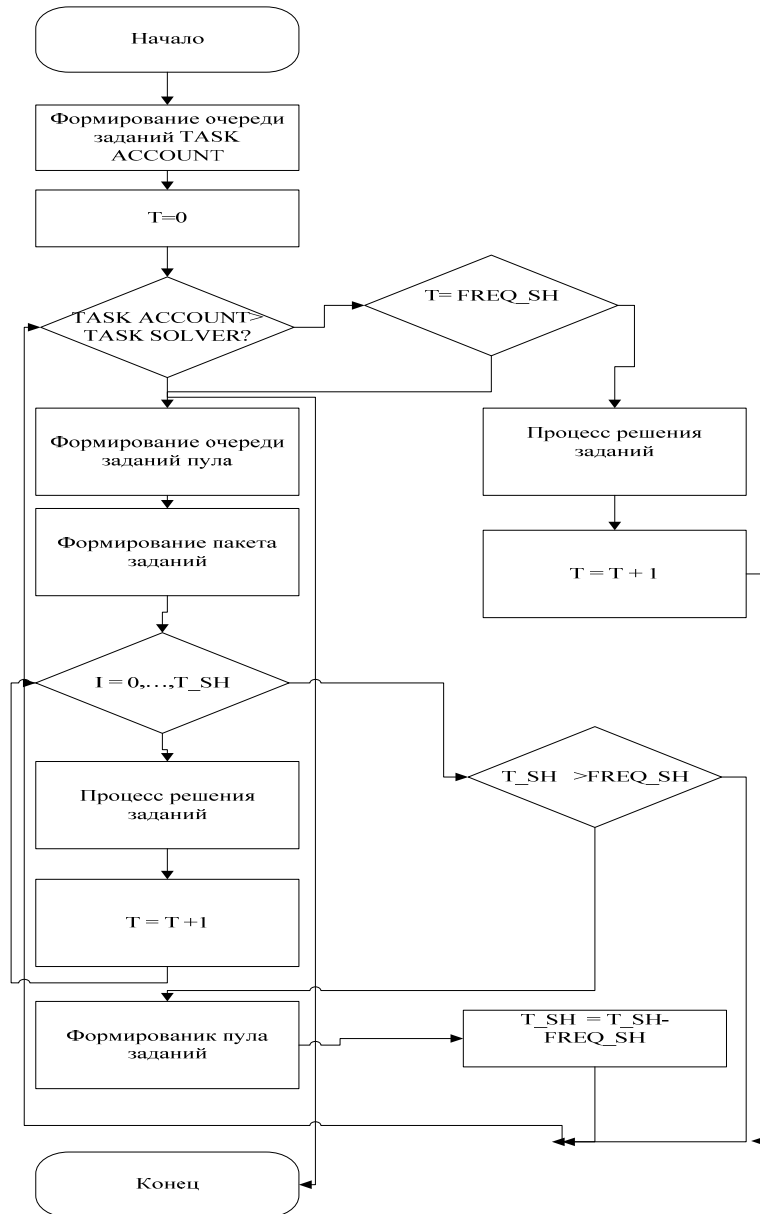


Рис. 1. Схема алгоритма работы модели

На схеме рис. 1 использованы следующие обозначения: TASK_COUNT – количество заданий, которые будут поданы на вход системы за все время моделирования; TASK_SOLVER – количество решенных заданий; T – такт работы системы; T_SH – время (в тактах), затраченное на формирование пакета заданий; FREQ_SH – период формирования пакета заданий (период планирования).

В предлагаемой модели используются следующие характеристики системы (табл. 2).

Таблица 2. Состав и описание характеристик системы

№ п/п	Наименование характеристики	Описание	Закон распределения
1	Количество заданий	Количество заданий, которое будут подано на вход модели для выполнения	Нормальный, Пуассона, Эрланга, равномерный
2	Количество ресурсов	Количество всех разных ресурсов, которыми решаются задания глобальной очереди	Нормальный, Пуассона, Эрланга, равномерный
3	Сложность решения задания	Максимальное количество тактов, за которое задание будет решено на одном ресурсе производительностью 1 такт	Нормальный, Пуассона, Эрланга, равномерный
4	Универсальность задания	Количество ресурсов, которыми может быть решено задание, в процентах к общему количеству ресурсов системы	Постоянное значение
5	Уникальные задания	Отношение количества заданий, которые решаются на одном ресурсе, к общему количеству заданий пула	Постоянное значение
6	Пул заданий	Буфер системы, в который загружаются все задания глобальной очереди для дальнейшего их распределения между ресурсами	Экспоненциальный, нормальный, Пуассона, постоянное значение
7	Производительность ресурса	Максимальное количество тактов «сложности решения» задания, которое ресурс может выполнить за один такт	Нормальный, Пуассона, Эрланга, равномерный
8	Пакет заданий	Количество заданий, которое может сформировано на выбранный ресурс	Постоянное значение
9	Коммуникационная задержка	Количество тактов, которое характеризует временную задержку передачи задания на выбранный для ее решения ресурс	Постоянное значение

Задания и ресурсы, на которых они решаются, задаются в виде прямоугольной таблицы (матрицы), строки которой соответствуют номерам заданий, а столбцы – номерам ресурсов, причем, если значение в ячейке m_{ij} устанавливается 1, это означает, что i -е задание решается j -м ресурсом (рис. 2).

В качестве метода для планирования ресурсов в данной работе используется задача о наименьшем покрытии (ЗНП), которая представляет собой задачу линейного булевого программирования, математическая постановка которой предлагается в таком виде:

$$L_i = \sum_{j=1}^n x_j(t_k) \rightarrow \min \quad (1)$$

при ограничениях

$$\begin{aligned} \sum_{j=1}^n \beta_{ij} x_j(t_k) &\geq 1, \quad i = \overline{1, m}; \\ \beta_{ij} &\in \{0, 1\}; \quad x_j(t_k) \in \{0, 1\}. \end{aligned} \quad (2)$$

где m – количество заданий, подлежащих планированию в момент времени t_k ; n – количество ресурсов исследуемой системы, доступных и свободных на момент планирования t_k ; $t_k \in [T_0, T_N]$ – интервал времени между началом и окончанием процесса планирования.

После формирования пула заданий выполняется операция назначения заданий пула на ресурсы в соответствии с выбранным методом планирования (табл. 3).

Особенностью предлагаемого метода является то, что для каждого задания находится несколько возможных ресурсов для ее решения, а так заданий множество, выбор ресурса определяется на основе матрицы на рис. 2.

Всего: 0 На такте: 0 Возвращенные: 0

№	Задача №	Позиция задач в пуле									
		1(0)	2(0)	3(0)	4(0)	5(0)	6(0)	7(0)	8(0)	9(0)	10(0)
1	1	1						1	1		
2	2				1					1	
3	3				1	1					
4	4	1			1		1	1			
5	5			1		1				1	
6	6					1				1	
7	7				1		1	1			
8	8				1		1	1			
9	9			1		1			1	1	
10	10		1		1	1		1			
11	11						1				1
12	12			1	1	1	1				
13	13					1		1			
14	14		1						1		
15	15			1	1	1		1			
16	16				1	1	1				
17	17		1	1				1			1
18	18				1	1					1
19	Значение «1» в ячейке таблицы, означает, что задача с номером строки, решается ресурсом с номером столбца										
20											

Рис. 2. Матрица соответствия заданий ресурсам

Таблица 3. Состав используемых методов планирования

№ п/п	Метод планирования	Алгоритм	Выбор ресурса
1	First Come First Served (FCFS)	Обслуживается первое задание из пула заданий	Выбирается первый свободный ресурс из доступных на момент планирования ресурсов
2	Shot Job First (SJF)	Обслуживается самое короткое задание из пула заданий	Выбирается первый свободный ресурс из доступных ресурсов на момент планирования ресурсов
3	Random	Выбор задания из очереди осуществляется по случайному закону	Выбирается первый свободный ресурс из доступных и свободных на момент планирования ресурсов
4	Minimal Cover (MC)	Назначение заданий на ресурсы определяется в соответствии с методом решения задачи о наименьшем покрытии	Выбираются для назначения на них заданий те ресурсы, которые определяются как результат решения задачи о покрытии, причем, если таковых несколько, ресурс выбирается по принципу первый свободный из доступных и свободных на момент планирования ресурсов

Для сравнительного анализа поведения предлагаемых методов планирования в модели были использованы следующие:

метод, использующий для планирования задачу о наименьшем покрытии (MC): метод находит наименьшее количество ресурсов (столбцов матрицы соответствия), которые могут решить все задания (строки матрицы), находящиеся в пуле. После этого задания распределяются на те ресурсы, которые вошли в наименьшее покрытие, и, если они загружены полностью, а задания в пуле еще остались, то они распределяются на другие свободные ресурсы. Оставшиеся после этого нераспределенными задания возвращаются обратно в пул, после чего пул вновь заполняется заданиями из глобальной очереди заданий и приведенная процедура повторяется.

В данном методе реализовано 2 режима работы: с фиксированным размером пула и переменным, который отражает интенсивность поступления заданий в соответствии с заданными законами распределения, отражающие реальные загрузки кластеров Грид.

Модификации метода FCFS:

FCFS без пакета заданий – метод распределяет задания по принципу «первый пришел первым обслужился», то есть в порядке поступления заданий или в порядке расположения их в очереди пула. Планировщик берет первое задание из пула и пробует разместить его на свободный ресурс, которым оно решается. Если имеется хотя бы один свободный для выполнения этого задания ресурс, то осуществляется переход к новому заданию, если нет, – задание возвращается в пул и переходит к следующему заданию в очереди.

FCFS с пакетом заданий – метод распределяет задания по принципу «первый пришел первым обслужился», то есть в порядке поступления заданий или в порядке нахождения их в очереди. Планировщик берет первое задание из пула и пробует разместить его в пакет заданий на тот ресурс, на котором оно решается. Если находится хотя бы один свободный для этого задания ресурс, то планировщик переходит к новому заданию пула, если нет, – задание возвращается в пул и осуществляется переход к следующему заданию в очереди.

Имитация процесса решения заданий на выбранных (назначенных) ресурсах заключается в следующем. каждому ресурсу в определенном порядке присваивается индивидуальный номер. Этот номер соответствует номеру ресурса, на котором может быть решено задание. Для ресурсов устанавливается длина (размер) пакета заданий, в который помещаются задания, которые решаются на данном ресурсе. Каждому заданию определен показатель «сложность решения» в тактах, который определяет, за какое количество тактов задание будет решено на ресурсе с производительностью один такт, а для каждого ресурса задается показатель «производительность», который также измеряется в тактах. Прежде чем назначенное на ресурс задание начнет решаться, выполняется коммуникационная задержка, которая имитирует задержку поступления заданий для их выполнения на ресурсе (передача данных, передача задания, системные опросы).

Результаты работы модели анализируются в соответствии со следующими метриками производительности Грид-системы (табл. 4).

Таблица 4. Метрики производительности работы Грид-системы

№ п/п	Наименование метрики	Описание метрики	Единица измерения
1	Общее время выполнения заданий глобальной очереди	Время решения всех заданий очереди с момента начала работы системы до момента решения последнего задания из глобальной очереди	Такт
2	Среднее время выполнения задания	Среднее время выполнения системой одного задания, начиная с момента поступления задания на вход системы (помещения в пул) и заканчивая временем решения задания на выбранном ресурсе	Такт
3	Время нахождения задания в пуле	Среднее время нахождения задания в очереди пула	Такт
4	Время нахождения задания в пакете заданий на ресурс	Среднее время нахождения задания в пакете заданий на назначенный для его решения ресурс	Такт
5	Коэффициент использования ресурса*100%	Среднее геометрическое время работы одного ресурса относительно всего времени работы системы (выполнения заданий глобальной очереди)	%
6	Коэффициент загрузки*100%	Среднее геометрическое количества заданий, которое было решено на одном ресурсе, относительно общего количества заданий глобальной очереди	%
7	Коэффициент ускорения*100%	Отношение суммарной сложности решения заданий глобальной очереди ко времени, затраченному на выполнение всех заданий глобальной очереди	%

Программная реализация имитационной модели

Программа модели реализована на основе оболочки программирования C++ Builder.

Для программной реализации предлагаемой модели разработаны диаграмма классов и интерфейсов (рис. 3) и диаграмма последовательностей (рис. 4).

В данном исследовании проведен сравнительный анализ работы методов планирования ресурсов на основе метода наименьшего покрытия (MC) и метода FCFS, который наиболее распространен в современных планировщиках.

Имитационное моделирование работы планировщиков на основе разработанной программы симуляции работы модели GRID_Scheduler_Model включает выбор и установку следующих параметров системы:

количество заданий в очереди (было выбрано среднее количество заданий за неделю по кластеру ANL-Intrepid [9], что составило 2100), равномерный закон соответствия заданий ресурсам. Данный параметр позволяет определить плотность распределения для матрицы соответствия задания-ресурсы (в данной работе рассмотрен вариант использования равномерного закона);

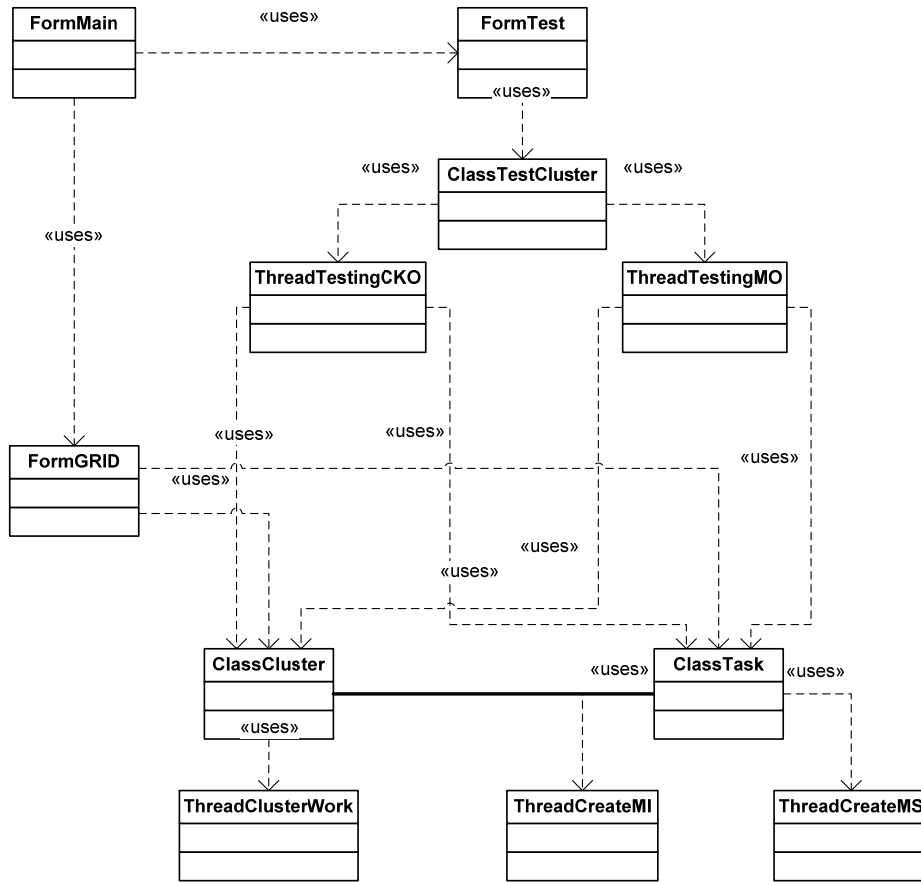


Рис. 3. Диаграмма классов и интерфейсов программной реализации модели

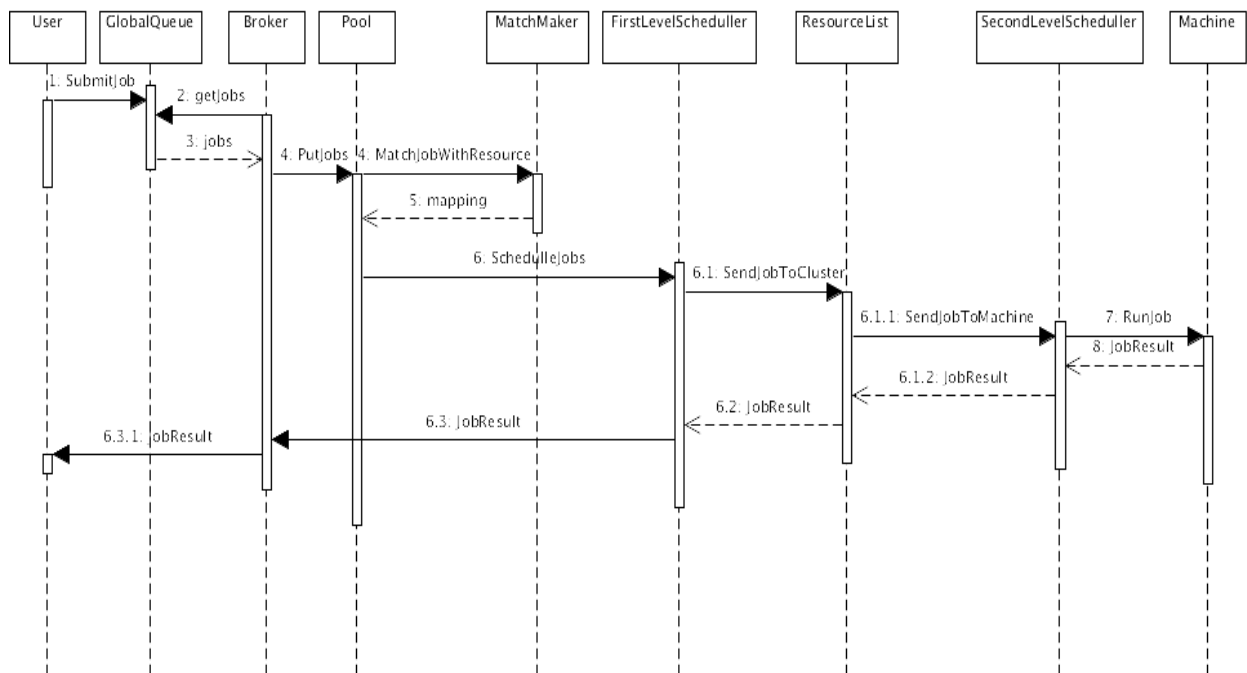


Рис. 4. Диаграмма последовательностей программной реализации модели

сложность решения заданий. Параметр определяет количество тактов для решения задания на одном ресурсе. В качестве среднего значения было выбрано 20 тактов, что соответствует сложности 20000 MI (million instructions согласно спецификации SPEC [10]). Данное значение было определено на основе статистического анализа сложности решения заданий на кластере ANL-Intrepid. В качестве закона распределения для сложности заданий был выбран закон Пуассона;

универсальность заданий. Параметр определяет процент (%) заданий, которые могут быть решены на любом из имеющихся ресурсов. При проведении эксперимента было определено, что 50 % из общего количества являются универсальными заданиями;

длина пула определяет количество заданий, которые поступают на планирование из глобальной очереди. В данном эксперименте это количество заданий определялось по нормальному закону (рис. 6), для которого среднее значение выбрано равным 300. Среднее значение было определено на основе результатов статистической обработки данных поступления заданий за сутки на кластер ANL-Intrepid за период, который равнялся 233 дням.

На рис. 3 использованы классы и интерфейсы программной реализации модели, состав и идентификаторы которых показаны в табл. 5.

Таблица 5. Состав классов и интерфейсов программной реализации имитационной модели

Наименование класса	Идентификатор
Класс интерфейса главного окна программы	FormMain
Класс интерфейса окна моделирования работы Грид	FormGRID
Класс интерфейса создания тестов работы Грид	FormTest -
Класс заданий	ClassTask
Класс потока, создающего таблицу соответствий между заданиями и ресурсами	ThreadCreateMI
Класс потока, создающего сложность решения заданий	ThreadCreateMS
Класс модели кластера Грид	ClassCluster
Класс потока, моделирующего работу кластера Грид	ThreadWorkCluster
Класс тестирования Грид	ClassTestCluster
Класс потока, вычисляющего математическое ожидание метрик производительности работы Грид	ThreadTestingMO
Класс потока, вычисляющего среднеквадратические отклонения метрик работы Грид	ThreadTestingCKO

На рис. 5 показана сгенерированная матрица соответствия для 2100 заданий и 100 ресурсов (в скобках для заданий указана сложность, для ресурсов – производительность в тактах).

№	Задание №	93(5)	94(3)	95(4)	96(3)	97(4)	98(4)	99(4)	100(4)
2088	2088(13)	1	1		1	1	1		1
2089	2089(12)	1	1			1			
2090	2090(10)				1				
2091	2091(10)		1					1	
2092	2092(14)				1	1	1	1	
2093	2093(8)								
2094	2094(6)	1		1		1			1
2095	2095(14)					1	1	1	
2096	2096(9)					1			1
2097	2097(4)		1	1	1	1			
2098	2098(8)	1		1		1			
2099	2099(10)					1			
2100	2100(6)	1		1					

Рис. 5. Матрица соответствия заданий ресурсам

В эксперименте использовано 100 ресурсов, производительность каждого из них – 5 тактов, что соответствует 5000 MIPS согласно спецификации SPEC. Данная характеристика близка к производительности современного 2-х ядерного процессора. Таким образом, общая производительность всех ресурсов равняется 500000 MIPS.

Размер пакета заданий для планирования на ресурсы определялся только для эвристического алгоритма планирования MC – на основе решения задачи о наименьшем покрытии (для метода FCFS не использовалось пакетное планирование). В эксперименте размер пакета был выбран равным количеству ресурсов (рис. 6).

Коммуникационная составляющая модели определяется параметром задержки. Данная характеристика позволяет учесть время передачи данных и заданий на ресурсы (было определено среднее время для заданий в размере одного такта – 1000 MI).

В результате имитационного моделирования работы разных методов планирования при использовании выбранных параметров модели были получены следующие результаты (рис. 7 – 9).

Эвристический алгоритм планирования MC позволил почти в шесть раз быстрее решить все задания глобальной очереди по сравнению с методом FCFS (3582 против 21319 тактов).

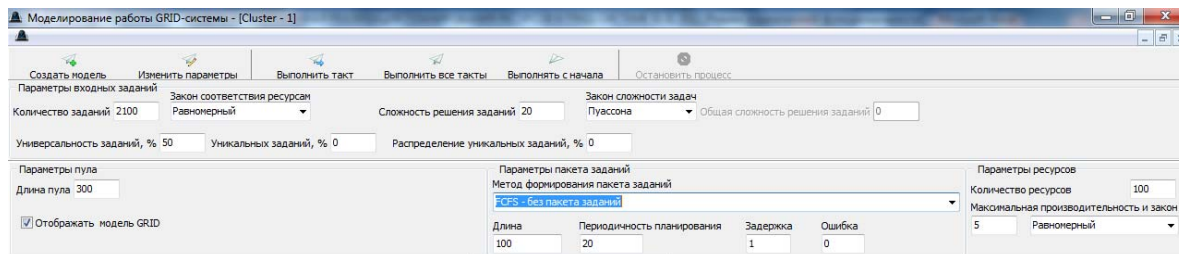


Рис. 6. Настройка параметров характеристик заданий, пакета заданий, ресурсов и периодичности планирования

На рис. 7 (в табличном виде) и рис. 8 (графическом виде) показаны характеристики времени выполнения всех заданий для двух рассмотренных методов планирования. При этом соотношение выигрыша метода MC сохраняется и для среднего времени выполнения задания в системе (2351 против 13940 тактов).

Наиболее показательным для сравнительного анализа методов является коэффициент использования ресурсов системы. В случае применения эвристического алгоритма MC он составляет 1,3%, в случае же использования FCFS – 0,4 %. Низкие значения коэффициента использования всех ресурсов объясняются тем, что при низкой интенсивности и средней универсальности (50%) заданий планировщик не может загрузить все ресурсы полностью на всех тактах. На рис. 8, 9 показано, что алгоритм MC учитывает гетерогенность ресурсов и заданий и поэтому более плотно «упаковывает» ресурсы заданиями, что позволяет получить выигрыш коэффициента использования для всех ресурсов. Рассчитанные коэффициенты использования отдельных ресурсов для выбранных методов планирования значительно отличаются: метод MC заранее планирует задания на соответствующие ресурсы (поэтому наиболее производительные ресурсы или соответствующие уникальным заданиям типы ресурсов загружены больше), а метод FCFS ставит задания на любой освободившийся и подходящий для их решения ресурс.

Проведенные эксперименты показали, что эффективность планировщика на основе метода MC можно повысить еще на 50–70% за счет выбора размера пакета заданий, периодичности планирования для выбранных характеристик законов распределения заданий (среднее значение количества заданий за сутки).

Отображение: подуглы модели кластера	Время: характеристики	Коэффициенты ресурсов	Список результатов работы кластера		
	MC1	MC2	MC3	FCFS	Создан GRID со следующими параметрами: 1. Количество ресурсов: 100 2. Количество заданий: 2100 3. Размер пула: 300 4. Размер пакета заданий: 100 5. Универсальность заданий: 50 6. Процент уникальных заданий: 0 7. Распределение уникальных заданий: 0 8. Максимальная сложность решения заданий: 20 9. Общая сложность решения заданий: 20991 10. Закон распределения уникальностей заданий: Равномерный 11. Закон распределения сложности решения заданий: Пуассона
Время выполнения задач	3582			21319	
Время ответа	2351,3119047619			13940,2338095238	
Время ожидания	2100,74714285714			12934,1071428571	
Время обслуживания	244,26619047619			0	
Время формирования пакета заданий	133,136363636364			1632,28947368421	
Коэффициент использования ресурсов	0,0131744			0,0043697	
Коэффициент загрузки ресурсов	0,0049242			0,0099999	
Коэффициент укорочения	2,9300671			0,4923074	

Рис. 7. Таблица метрик производительности методов планирования

Таким образом, результаты моделирования с использованием различных методов планирования показали, что планировщик на основе предложенного метода планирования MC в несколько раз превосходит широко используемый алгоритм FCFS по значениям всех метрик производительности работы модели Грид-системы. Это доказывает целесообразность и эффективность его применения для Грид-систем для повышения их производительности в процессе эксплуатации. Предложенный в данной работе метод позволяет решить задачи повышения производительности расписаний выполнения заданий и загрузки ресурсов Грид-системы. При этом важным является определение возможных соотношений между значениями метрик системы, которые позволят анализировать ее поведение в условиях изменяющихся количества пользователей, интенсивностей потоков заданий и доступных для вычислений ресурсов.

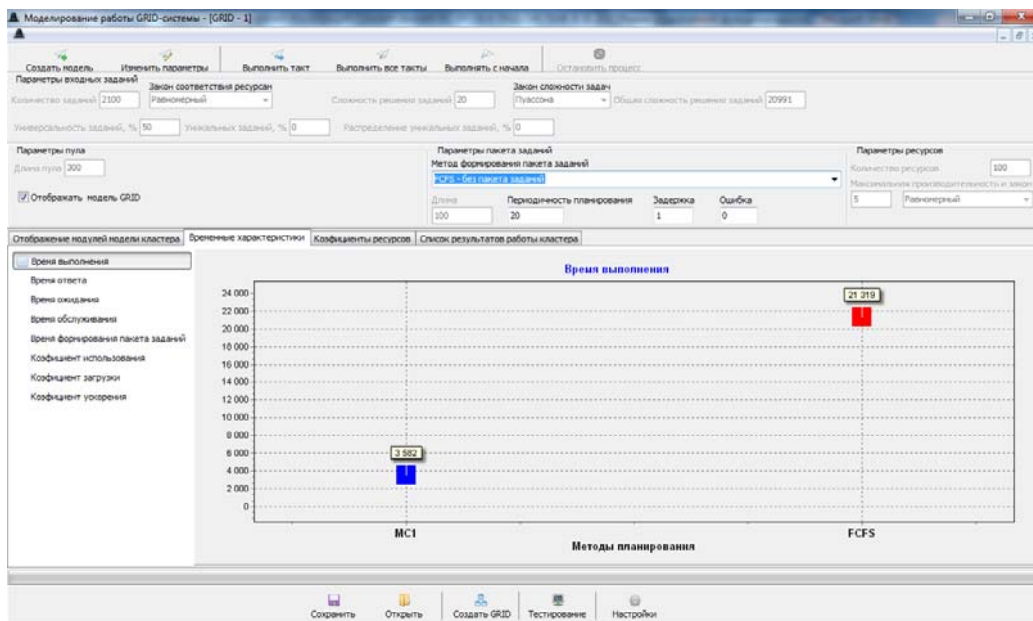


Рис. 8. Анализ времени выполнения заданий глобальной очереди для выбранных методов планирования

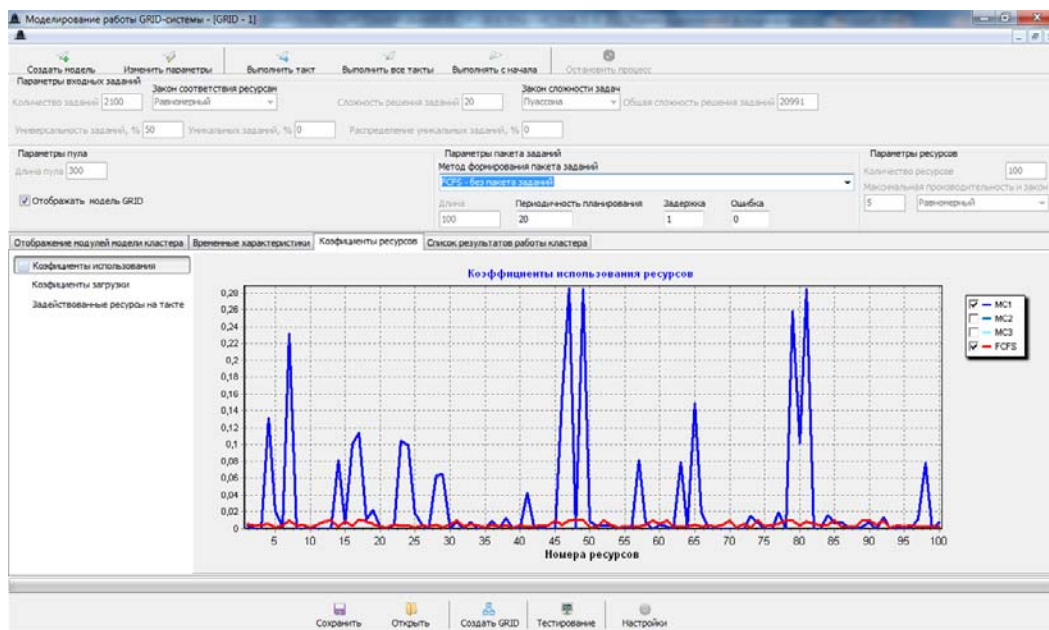


Рис. 9. Анализ коэффициента использования каждого ресурса для выбранных методов планирования

1. GSSIM. [Электронный ресурс] – Режим доступа <http://www.gssim.org>.
2. OMNeT++. [Электронный ресурс] – Режим доступа <http://omnetpp.org>.
3. OptorSim. [Электронный ресурс] – Режим доступа <http://edg-wp2.web.cern.ch/edg-wp2/optimization/optorsim.html>.
4. MicroGrid. [Электронный ресурс] – Режим доступа <http://wwwcsag.ucsd.edu/projects/grid/microgrid.html>.
5. GridSim. [Электронный ресурс] – Режим доступа <http://www.gridbus.org/gridsim>.
6. SimGrid. [Электронный ресурс] – Режим доступа <http://gcl.ucsd.edu/simgrid..>
7. Петренко А.І. Комп'ютерне моделювання грід-систем // Електроніка і зв'язь 5' Тематический випуск «Електроніка і нанотехнології», 2010. – С. 40–48.
8. Минухін С.В., Коровин А.В. Моделирование планирования ресурсов GRID средствами пакета GridSim // Системи обробки інформації. Інформаційні технології та комп'ютерна інженерія. – 2011. – Вип. 3(93). – С. 62 – 68.
9. http://www.cs.huji.ac.il/labs/parallel/workload/1_anl_int/index.html.
10. Standard Performance Evaluation Corporation [Электронный ресурс] – Режим доступа <http://www.spec.org>.